CHAPTER ONE

Some Principles Underlying the Cognitive Approach to Instructional Design*

Wellesley R. Foshay

In the generation since the birth of the instructional design field, our understanding of the basic psychological mechanisms of memory, perception, learning, and problem solving has seen a great deal of development. Corresponding progress in our understanding of the psychology of instruction (or, if you prefer, design of learning environments) has led to important new definitions of principles of instructional design. For those familiar with the behavioral approach, this chapter will review what you already know and show how the cognitive approach differs. For those who have never had a formal study of the assumptions underlying the behavioral approach, this chapter will provide you with a theoretical understanding of the approach you probably have been using to date. Important additional principles are included in the chapters in Part Three. However, a full discussion of the psychology of learning and instruction is beyond the scope of this chapter and of this handbook. If you are interested in pursuing the subject matter further, references to sources from which this chapter is drawn are provided.

We do not mean to imply a disjunctive contrast between the behavioral and the cognitive approach, nor do we mean to imply that behavioral principles are obsolete—only that the cognitive approach often adds prescriptive utility to

* Portions of this chapter are adapted from Wellesley R. Foshay, Kenneth Silber, and Michael Stelnicki (2003), *Writing Training That Works: How to Train* Anyone *to Do Anything: A Practical Guide for Trainers Based on Current Cognitive Psychology and Instructional Design Theory and Research.* San Francisco: Pfeiffer.

our practice over a wide range of training needs. Few instructional designers follow a purely behavioral or cognitive approach to design. Furthermore, in many cases the behavioral approach and the cognitive approach lead to similar design solutions. Therefore you may find that you are already using some principles of the cognitive approach in designing your instruction.

HOW THE BEHAVIORAL APPROACH IS DIFFERENT FROM THE COGNITIVE APPROACH

Generally speaking, behaviorism is a set of principles concerning both human and non-human behavior. One major behaviorist goal is to explain and predict observable behavior. Behaviorists define learning as the acquisition of new behavior as evidenced by changes in overt behavior. Behaviorism draws conclusions about behavior from research on external events: stimuli, effects, responses, learning history, and reinforcement. These behaviors are studied and observed in the environment and are explained with little or no reference to internal mental processing.

In dramatic contrast to behaviorism, a major tenet of cognitive psychology is that internal thought processes cause behavior. It is their understanding that can best explain human behavior. Cognitive information processing psychologists consider learning to be mental operations that include internally attending to (perceiving), encoding and structuring, and storing incoming information. Cognitive psychologists interpret external stimuli in terms of the way they are processed. They use observable behavior to make inferences about the mind. Furthermore, exciting new work in cognitive neuroscience is relating the structure of the brain to its function, and in the process, validating and elaborating on the accounts of processing and memory induced experimentally by the cognitive psychologists.

The difference in focus between the behaviorist and cognitive theories has important implications for instructional designers who seek design principles based on theory. The biggest differences are in these theoretical areas:

- What learning is
- Factors influencing learning
- The role of memory and prior knowledge
- How transfer occurs
- The goal of instruction
- The structure of instruction
- Specific instructional strategies

Different types of learning are best explained by each approach, and each approach provides basic principles that guide instructional design in different circumstances.

What the implications are for each of the above areas and how they differ in each of the two approaches are shown in Table 1.1. It is important to note that some of the differences are merely semantic (for example, ''fluency'' and ''automaticity'' both describe degrees of learning proficiency), while some are more substantive. For example, ''emphasis on knowledge structures'' reflects the cognitive theory's recognition of the need to think about the parts of knowledge in any given subject and how they fit together.

Table 1.1 Differences Between Behavioral and Cognitive Approaches

Instructional Design Area	*Behavioral Approach*	*Cognitive Approach*
What learning is	''changes in form or frequency of observable performance''; what learners ***do***	internal coding and structuring of new information by the learner; discrete changes in knowledge structures; what learners ***know*** and ***how*** they come to know it
Factors that influence learning	''arrangement of stimuli and consequences in the environment''; reinforcement history; fluency in responding	how learners attend to, organize, code, store, retrieve information as influenced by the context in which information is presented when it is learned and when it is used; thoughts, beliefs, attitudes and values; automatic responding
The role of memory	not addressed in detail; function of the person's reinforcement history; forgetting results from lack of use	''learning occurs when information is stored in memory in a meaningful manner so it can be retrieved when needed''; ''forgetting is the inability to retrieve information from memory because of interference, memory loss,

(*Continued*)

Table 1.1 (*Continued*)

Instructional Design Area	*Behavioral Approach*	*Cognitive Approach*
		or inadequate cues to access the information'' given the way it is organized in memory; therefore, meaningfulness of learning directly affects forgetting
How transfer occurs	focus on design of the environment; stimulus and response generalization to new situations	stress on efficient processing strategies to optimize cognitive load; function of how information is indexed and stored in memory based on expected use of the knowledge; applying knowledge in different contexts by reasoning analogically from previous experiences; construction/ manipulation of mental models made up of networks of concepts and principles; learners believe knowledge is or will be useful in new situation
What types of learning are best explained by the approach	discriminations (recalling facts); generalizations (defining and illustrating concepts); associations (applying explanations); chaining (automatically performing a specified procedure)	''complex forms of learning (reasoning, problem solving, especially in ill-structured situations)''; generalization of complex forms of learning to new situations
What basic principles of the approach are relevant to ID	produce observable, measurable outcomes => task analysis, behavioral objectives, criterion-referenced testing; existing	All of the behavioral principles, and: student's existing mental structures => learner analysis; guide and support for accurate

	response repertoire and appropriate reinforcers => learner analysis; mastery of early steps before progressing to complex performance => simple to complex sequencing; practice; mastery learning; reinforcement => practice. followed by immediate feedback and rewards; use of cues and shaping => prompting, fading, sequencing	mental connections => feedback; learner involvement in the learning process => learner control; metacognitive training; collaborative learning; identify relationships among concepts/principles to be learned, and between them and learners' existing mental models => learner analysis; cognitive task analysis; emphasis on structuring, organizing and sequencing information for optimal processing => advance organizers, outlining, summaries; connections with existing knowledge structures through reflective processing => analogies, relevant examples, metaphors
Goal of instruction	elicit desired response from learner presented with target stimulus	make knowledge meaningful and help learners organize and relate new information to existing knowledge in memory
How should instruction be structured	determine which cues can elicit the desired responses; arrange practice situations in which prompts are paired with target stimuli that will elicit responses on the job; arrange environmental conditions so students can make correct responses in the presence of target stimuli and receive reinforcement	determine how learners' existing knowledge is organized; determine how to structure new information to mesh with learners' current knowledge structure(s); connect new information with existing in meaningful way through analogies, framing, outlines, mnemonics, advance organizers; arrange practice with structurally meaningful

(*Continued*)

Table 1.1 (*Continued*)

Instructional Design Area	*Behavioral Approach*	*Cognitive Approach*
		feedback so new information is added to learners' existing knowledge
Specific instructional strategies	teach fact lesson first, then concepts, then principles, then problem solving; focus on algorithmic procedures for problem solving, including troubleshooting; teach each concept, procedural chain, troubleshooting approach separately; when mastered go on to next; focus on deductive learning; present principles and attributes; build generalization with extended realistic practice, often after initial acquisition	teach problem solving in authentic (job) context; teach principles, concepts, and facts in context as appropriate within the problem-solving lesson; focus on heuristic problem solving and generalization, even in troubleshooting; teach overall mental model, then use coordinate concept, principle, procedure/problem solving teaching to teach all related knowledge at or near the same time; focus on inductive learning; present examples; build generalization through practice in additional problems and contexts which require similar but not identical problem-solving procedures

Based on J. R. Anderson, 1995a; J. R. Anderson, 1995b; Ertmer & Newby, 1993; Fleming & Bednar, 1993; Foshay, 1991; Hannafin & Hooper, 1993; Silber, 1998; West, Farmer, & Wolff, 1991

WHY THE COGNITIVE APPROACH TO INSTRUCTIONAL DESIGN IS IMPORTANT

The cognitive approach to ID has become prominent in the past two decades for two reasons, one based in the theory of learning and instructional design, the other based in business. From the perspective of theory, the cognitive approach seeks to overcome a number of limitations of the behavioral approach. For example, with the behavioral approach to ID:

- Learners sometimes have trouble transferring what they have learned from training to the job;
- Learners can have trouble attaining expert-level performance in troubleshooting and problem solving on the job;
- Learners often have trouble generalizing their training from one situation to another, leading to skill gaps every time the job, content, or technology changes, and creating the need for retraining;
- Learners may have difficulty with divergent reasoning (many right answers or many ways to get to the answer), as opposed to convergent reasoning (one right answer and one way to get it); and
- Designers do not have adequate prescriptions for designing the kinds of training we are now being asked to design—problem solving, troubleshooting (especially in settings where content volatility is high), design, heuristic-based thinking (using guidelines versus algorithmic thinking, which uses formulas with 100 percent predictable outcomes), strategic thinking, and the like.

From the perspective of business, the current behavioral approach to ID sometimes leads to excessive development and delivery costs because it requires:

- Longer training sessions, to cover all the specific algorithms or other content variations;
- More retraining time, to address lack of transfer to new situations; and
- More development time, because there are no guidelines for creating training for higher-order thinking, developers must either guess, or treat problem solving as a large number of low-level procedures and concepts.

The cognitive approach to ID offers remedies to these problems. It provides designers with another way to design training that works well in situations in which higher-order thinking, problem-solving, and transfer to new situations are training goals.

HOW LEARNING OCCURS ACCORDING TO THE COGNITIVE POINT OF VIEW

There are many theoretical models in cognitive psychology. Although there are important differences among them, they broadly agree on how learning occurs. According to these models, there are several components of the mind, and each is involved in the learning process in certain ways. How each component of the mind works has implications for how we design instruction. The components are

- Perception and memory stores
- Short-term or working memory
- Long-term memory

Perception and Memory Stores

Perception Is Selective. There is more stimulation in the environment than we are capable of attending to, and then encoding (internally translating) for storage in memory. Therefore, we only attend to certain things. We attend to and see/hear what we expect to see in a given situation. We attend to those things that interest us because they are either (a) related to what we already know or (b) so novel they force us to attend to them.

Limits of the Sensory Stores. Our sensory stores, also called sensory memories (analogous to a computer's ''buffers''), are capable of storing almost complete records of what we attend to. The catch is they hold those records *very briefly*. During that very brief time before the record decays, we do one of two things: (1) we note the relationships among the elements in the record and encode it into a more permanent memory or (2) we lose the record forever.

ID Implications. The implications of the selectiveness of perception and limitations of sensory stores for instructional designers are that it is crucial to:

- Get the learner to *attend to* the parts of the environment that are crucial (hence the emphasis in the cognitive approach to ID on attention-getting and on motivational statements); and
- Help the learner note *relationships* among the information quickly (hence the importance of organizing the information you are presenting and of clearly relating the new information to existing familiar or important contexts and knowledge).

Short-Term or Working Memory

Controversy. There is disagreement among cognitive psychologists about whether there is a short-term memory that is ''separate and different'' from long-term memory. The disagreement is about whether the two types of memory are physically different, or whether they are just conceptually different constructs. There is also discussion about how they encode information, how they store information, and so on. Regardless of the theoretical differences, some ideas that most psychologists would agree about can affect the design of training.

Rehearsal. When information is passed from the sensory stores to memory, we mentally rehearse it. Examples include repeating phone numbers several times or creating associations to names (for example, **Ted** with the **red** hair) to help memorize them when you first hear them at a party. The former, simply repeating the information over and over, is called *passive rehearsal*. It does not seem to improve memory as well as rehearsing the information in a *deep and meaningful* way, like the latter way of creating associations.

Limited Capacity. There seems to be a limit on the amount of information we can rehearse at one time. A classic paper presented by a Bell Labs psychologist in 1956 showed that we can remember 7 +/− 2 bits of information at most, and that to remember more we have to ''chunk'' (or group) information in manageable sizes; that's why your phone number has seven digits, and when area codes became prominent, people were taught to remember phone numbers in three chunks (aaa-bbb-cccc). The findings of this study still seem to apply, with some modifications of how you define a ''bit'' (element) or a ''chunk'' (and, as you will see later in Part Four, the ''7 +/− 2'' estimate is probably too high in many circumstances).

Format. At this point in the learning process, the information being rehearsed is not yet organized and encoded as it will be when it is finally stored in memory. Also, there is some evidence that there are separate spaces for storing and rehearsing verbal information and visual/spatial information, and possibly separate spaces for other types of memories as well.

ID Implications. The implications are that instructional designers need to:

- Help learners use meaningful ways of rehearsing the information, as opposed to simply repeating it (through the use of analogies, by relating new information to existing knowledge or problem situations, etc.);
- Present the information in meaningful ''chunks'' of appropriate size for the learner population (knowing what your learners already know about the subject they are learning is critical to determining what ''appropriate size'' for those learners is);
- Present the information in multiple formats (verbal, auditory, visual), which can help learners rehearse, and therefore remember, better; and
- Present the information in a way that allows the learner to move quickly from rehearsing the information to encoding it and integrating (indexing) it with other information into long-term memory.

Long-Term Memory

In general, theorists believe that long-term memory is organized based on context and experience. That means we encode, store, and retrieve information in the way we have used knowledge in the past and expect to use it again in the future. There are several phenomena psychologists agree on about what strengthens the memory process.

Memory Strength. Information in memory has a characteristic called *strength*, which increases with practice. There is a *power law of learning* that governs the relationship between amount of practice and response time or error rates (Strength = Practice to power x). In simple terms, this means that practice increases the strength of learning exponentially (for example, double the practice at least squares the strength of the learned information in memory; triple the practice increases the strength by a factor of nine). Note that other factors, such as meaningfulness, also affect memory strength.

Elaboration. Elaboration means adding information to the information we are trying to learn. The more we elaborate on what we learn through processing, the better we remember it. This is because, as we tie the new information to existing information or as we create other information related to the new information, we create more pathways to get to the new information as we try to remember it.

Chunking. Memories are stored not as individual bits or as long strings of information, but in ''chunks,'' with each chunk containing about seven elements. As explained in the section above on short-term memory, how big an ''element'' and a ''chunk'' are differs based on the learner's existing knowledge.

Verbal and Visual Information. It seems we encode verbal and visual information differently in memory. We use a linear code for verbal information, and a spatial code for visual information. We remember visual information very well, especially if we can place a meaningful interpretation on the visuals. In addition, the Gestalt psychology finding that we remember incomplete and strange images better than complete, standard ones still appears to hold true. With verbal information, we remember the meaning of the information, not the exact words.

Associations and Hierarchy. Information is organized in memory, grouped in a set of relationships or structures (for example, hierarchically). Using such a structure makes it easier for us to remember, because there are more related pieces of information activated when we search for information. While you may not remember one specific piece of information in the structure, you may remember the overall structure and some pieces in it, and from that you can remember or infer the missing piece of information. For example, you may not remember all the numbers in the 12 × 12 multiplication tables, but if you remember some key ones (1, 2, 3, and 5 × a number) you can construct the rest.

By comparison with computers, humans can remember far fewer separate pieces of data, but are much better equipped for pattern recognition skills such as analogical reasoning, inference, and comprehension of visual and verbal languages.

ID Implications. The implications are that instructional designers need to:

- Build a lot of meaningful practice into training to increase the probability of retention (for example, the PQ4R method: Preview, Questions, Read, Reflect, Review, Recite);
- Provide learners with information (allowing them to create information that elaborates on the information to be learned);
- Present the information in meaningful ''chunks'' of appropriate size for the learner population (knowing your learners is critical);
- Present the information so it uses the abilities to remember both verbal and visual information, which can increase memory;
- Hierarchically organize the information being presented (to approximate the way information is stored in memory) to increase retention;
- Provide many associations to the information being learned to increase the chances that the information will be retrieved when called for;
- Help learners to organize/index their memories so they have many associations, many retrieval paths, and appropriate structures; and
- Use authentic (real-world) contexts for explanations, examples, and practice, which will help the learners relate what they learn to situations in which they will need to use the knowledge.

CATEGORIES OF KNOWLEDGE: DECLARATIVE AND PROCEDURAL KNOWLEDGE, AND THEIR SUB-TYPES

When they discuss learning, many cognitive psychologists draw distinctions among different categories of knowledge. When you design training, you will probably find it helpful to use these distinctions to help you decide what kind of knowledge you are teaching and how you can best teach that knowledge.

The biggest distinction is between *declarative* and *procedural* knowledge:

- *Declarative* knowledge is knowing *that*.
- *Procedural* knowledge is knowing *how*.

These are examples of *declarative knowledge*:

- Remembering your telephone number;
- Being able to tell the difference between a table and a tray; and
- Stating that for a car engine to run, it must have air, fuel, and electrical current for the ignition.

These are examples of *procedural knowledge*:

- Following a recipe to bake a cake;
- Building a spreadsheet ''from scratch'' using a software package for spreadsheets;
- Fixing the copier so it will stop jamming; and
- Designing a copier that can't jam.

The basic difference between the two types of knowledge is that declarative knowledge tells you *how the world is*, while procedural knowledge tells you *how to do things in the world*.

Trainers who don't understand this distinction often confuse *knowing* and *doing*, and thus make the following kinds of mistakes in designing training:

- They try to teach (and test) procedural knowledge using strategies suited for declarative knowledge;
- They teach declarative knowledge and stop, assuming that the procedural knowledge will naturally follow on its own; and
- They try to teach the procedural knowledge without teaching the associated declarative knowledge.

There are different types of declarative and procedural knowledge. It's important to understand them so that when you plan your instruction, you can use instructional strategies that are appropriate to each type. If you're good at making these distinctions, you may be able to save considerable time and expense in developing and delivering your training while improving its effectiveness. The different types are discussed below, and chapters in Part Three further explain how to teach each of the types of knowledge.

Types of Declarative Knowledge

One common practice is to distinguish three types of declarative knowledge:

- Facts
- Concepts
- Principles and mental models

The discussion below is a synthesis of much that is already familiar and commonly accepted on the topic; it is included here for completeness. The reader will note that these types of declarative knowledge are very similar to the types of learning proposed by Gagné and taught in most basic ID texts (such as Dick & Carey, 2001). To the traditional categories and explanations, however, we have added the notion of mental models, and described their characteristics in slightly different terms to align better with cognitive theory.

Facts. A fact is a simple association among a set of verbal and/or visual propositions. Some examples of facts are

- On a traffic light, red means stop, green means go, and yellow means prepare to stop;
- In 1492 Christopher Columbus sailed from Spain and landed in the Caribbean; he was not the first to do so, nor did he discover America;
- Miller's (Miller, 1956) studies for Bell Labs said the largest number of digits a person could remember easily was seven; and
- The five steps to create a table in MS Word 2003 for Windows are (1) select tables, (2) select number of rows, (3) select number of columns, (4) select line appearance, (5) click OK.

When you know a fact, you have placed it in a structure so you can recall it from memory. Learning facts as part of a structure that will help you recall them in the way you need them is much more efficient than trying to memorize each fact by itself. Simply knowing a fact does *not* mean you can use it in new situations, explain what it means, identify its relationship to other facts, or recall it to do anything with it.

Concepts. A concept is a category of objects, actions, or abstract ideas that you group together with a single name because they share characteristics in common. Some examples of concepts are

- Cars (versus trucks or campers or utility vehicles);
- Jogging (versus running, walking);
- Beautiful sunrises (versus beautiful sunsets, ugly sunrises);
- Justice (versus injustice); or
- Performance improvement (versus training).

When you know a concept, you can classify new objects, actions, or ideas as either in the category or not. People typically learn concepts by remembering the best example of the category they've seen (or imagined). They may or may not be able to articulate a verbal definition. Concepts do not exist in isolation; all concepts have related concepts (parts or kinds, more general, more specific). Items in a given category that do not belong to one concept in the category do belong to another concept in the category.

Principles and Mental Models. *Principles*. A principle is a cause-effect relationship. When you understand a principle, you know how something works. Principles are frequently stated as ''if . . . , then. . . . '' statements. You can demonstrate your understanding of a principle by explaining why something happens or predicting what will happen. For example, you know that:

If you see lightening nearby, you will hear thunder;

If you turn the ignition key in a car, the engine will probably start;

If you rob a bank, you may go to jail;

If you write test items to match instructional objectives, the test will have certain types of validity; and

E = IR (Ohm's law: electrical voltage is the product of current times resistance).

Mental Models. It's also important to know that the three types of declarative knowledge we've talked about so far fit together into structures. These structures are called *mental models*. They are networks of principles along with their supporting concepts and facts stored in a meaningful structure based on (a) the context for which it was created and (b) the past learning and experiences of the learner. For cognitive psychologists, mental models are the key to learning and using knowledge because:

- They tie together all the declarative knowledge in memory;
- They are the structures into which you organize information, put it into memory, retrieve it from memory when needed, and learn by expanding and restructuring existing structures;
- They provide the most meaningful application of declarative knowledge (as adults we rarely spout networks of facts or run around finding new instances of concepts, but we do frequently try to explain how or why things happen or work);
- They form a bridge between declarative knowledge (knowledge about) and procedural knowledge (knowing how) to do procedures (other than rote ones), you have to ''know how the system works,'' that is, have a mental model of the system; and
- The wrong ones (often called *misconceptions*) will actually interfere with performance and further learning.

Therefore, most would argue that for training of adults, the ID must not only teach isolated facts, concepts, and principles, but must also help the learner create the appropriate mental models for optimum structuring of the information learner for storage, retrieval, and application, while guarding against formation (or perpetuation) of misconceptions.

Types of Procedural Knowledge

Procedural knowledge is the ability to string together a series of mental and physical actions to achieve a goal. Procedural knowledge is used to solve problems.

The way ''problem'' is used in this book may be a new concept for many readers. In the behavioral approach, instructional designers are used to thinking about ''procedures'' and ''problem solving'' as two different things—two different levels in a hierarchy such as Gagné's. In the cognitive approach, and in this book, the tendency is to use ''procedural knowledge'' and ''problem solving'' interchangeably, which many might find confusing initially. Because procedural knowledge is used to solve problems, the type of problem the knowledge is used to solve is what leads to the name of the procedural knowledge.

Problem solving always has a starting or initial state (car not running), an end or goal state (running car), a sequence of actions (open door, get in, apply brake, insert key in ignition switch, turn key), and constraints (works only if you have the right key). These are summarized in Table 1.2.

Types of procedural knowledge and problem solving are placed on a continuum:

- At the most precise (procedural) end are well-structured problems;
- At the least precise end are ill-structured problems; and
- In the middle are moderately structured problems.

Table 1.3 summarizes the differences in problem types.

Well-Structured Problem Solving

A term you may sometimes hear for well-defined procedural knowledge is *rote procedure*. We consider performing rote procedures to be well-structured problem solving. All elements of the problem situation are known. The initial state, goal state, and constraints are clearly defined. The operations are also clearly defined, although they may include a choice of alternatives (branches). The learner knows when to start the procedure and when to stop it. Examples of well-structured problem solving include:

- Ringing up a sale in a department store;
- Calculating heating and air-conditioning requirements for a building;

Table 1.2 Problem Characteristics

Definition	*Example*
a. There is an **initial state**, or the elements of the problem the learner is presented with.	You want to record five different TV programs broadcast on five different nights each at a different time.
b. There is a **goal state**, or a description of the situation that would be a solution to the problem.	You need to program the VCR correctly to record the programs.
c. There is a **set of operations** or things the learner can do to get from the initial state to the goal state.	You need to follow the step-by-step programming procedure furnished by your VCR and TV set.
d. There is a **set of constraints** or conditions that must not be violated by the learner in solving the problem (Gagné, 1985). Anderson (1995) uses ''Search'' instead, as the mechanism of chaining together operations to get from initial to end state.	You must input the correct day, time, and channel for each program in the correct sequence. You must make sure there are no fund drives, presidential press conferences, ''special'' programs, or any other scheduling changes that would throw off the original times. You also have to make sure you're correctly specifying a.m. and p.m., correctly associated network name and channel number, and so on.

- Implementing a design for a database; and
- Printing marketing pieces.

Well-structured problems are usually performed simply by recalling procedures and performing them exactly as taught. It's not even necessary to understand

Table 1.3 How Problem Elements Differ in These Classes of Problems

	Well Structured	*Moderately Structured*	*Ill Structured*
Initial state	Clearly defined	Perhaps known	Not clear or spelled out
Goal state	Clearly defined	Clearly defined	Not clear or spelled out
Operations	Clearly defined	Must be created	Not clear or spelled out
Constraints	Clearly defined	May be known	Not clear or spelled out
Example	Access a computer file	Car won't start	No marketing plan for a new product

why the procedure works. Thus, in many situations it is optional to understand underlying *principles* that explain the *why* of a well-structured procedure.

Moderately Structured Problem Solving

In moderately structured problems, which include troubleshooting, the goal state is clear and the learners might know the initial state and constraints. However, the learners probably have to recall and assemble in a novel way the operations that will get them from the initial state to the goal state, given the constraints. Examples of moderately structured problems include:

- Fixing the cause of a ''mis-ring'' on a sale item in a department store;
- Developing a floor plan for a building;
- Planning how to implement a redesigned work process; and
- Planning a marketing focus group.

Other examples are deciding on the most advantageous retirement package, deciding whether or not to fire an employee, determining whether to repair your old laptop computer or buy a new one, determining whether or not to recommend that an employee seek company-provided counseling.

For moderately structured problems, it is important to understand the principles that underlie them. For example, a manager who wants to figure out how to motivate an employee needs to understand a few basic principles of motivation, if only at the common-sense level.

Troubleshooting (discussed further in Chapter Eleven) is a special ''compound'' case, in which an expert treats unfamiliar and/or complex problems as moderately structured and generates the operations. Some examples include determining the cause/source of a food poisoning outbreak, finding the source of a scraping noise when your car starts, determining why a metal stamping machine damages its stampings on a random basis, and figuring out why your refrigerator defrosts continually or your coffee maker doesn't work.

Ill-Structured Problem Solving

In ill-structured problems, which include most of the complex problems our learners encounter, three or all four of the elements of a problem are either missing completely or are present but not clear. The range of ill-structured problems is discussed further in Chapter Eleven. Examples of ill-structured problems include:

- Deciding on the sale price for an item in a department store;
- Designing a new building;

- Redesigning a work process; and
- Introducing a new product.

Other examples are holding a press conference on a highly controversial issue, conducting a workshop with learners who are highly resistant to learning the content, and designing an artificial pancreas or an acceptable human blood substitute or an automobile that never wears out.

You've probably heard the old saw that "defining a problem is most of solving it." That refers especially to these ill-structured problems.

The view of this book. For purposes of instructional design, in most circumstances there is little difference between moderately and ill-structured problems. Therefore, we will consider only two classes of problems: well-structured and ill-structured. Jonassen's treatment in Chapter Eleven is consistent with this view.

CONCLUSION: TWELVE INTERPRETIVE PRINCIPLES TO APPLY TO INSTRUCTIONAL DESIGN

Figuring out the implications of learning theory for instruction is neither direct nor simple. That's what instructional designers do, and it's what much of this book (especially Part Three) is about. To get you started, however, we will state twelve principles we hope will help you make the inferential leap from learning to instruction. Many of these are elaborated upon in Part Three.

Principle 1: Any Job Task Includes Both Declarative and Procedural Knowledge. It is naïve to think that entry-level jobs require rote procedures, and only higher-level jobs require workers to know both the "why" and the "how." If that was ever true, is certainly isn't in today's knowledge economy! Your front-end analysis, and your training, should always include both.

Principle 2: All Knowledge Is Learned in Structures, Which Are Related to the Logic of the Knowledge, What You Already Know, and How You Use the Knowledge. Stated differently, understanding the forest is as important as knowing the trees. You must think structurally about what is to be learned, and so must the learner.

Principle 3: There Are Different Types of Declarative Knowledge, and You Learn Each a Different Way. This insight is one of the cornerstones of instructional design. Your strategies, tools, and templates must vary by knowledge type, which means you must constantly be aware of what knowledge type(s) need to be learned to master a given task you are training.

Principle 4: Concepts and Principles Are Best Learned from a Combination of Examples and Definitions. A common design error we see is to treat examples as an afterthought, to fail to include them, or to include ones that don't work for the learning task and the learner. You and your learner need both examples and definitions, and both must be carefully constructed.

Principle 5: Teach the Knowledge Structure, Not Just the Parts in Isolation. You must help the learner build (or modify) appropriate knowledge structures. This is a separate learning task from understanding the pieces of knowledge separately, and you must provide opportunities for the learner to create and integrate these structures.

Principle 6: Procedural Knowledge Is How to Do. Anything that involves a series of steps is a procedure. Real-world job tasks usually include a combination of embedded procedures.

Principle 7: Procedural Knowledge Varies According to Its Structure. A common design error is to look only for the well-structured procedures, or to treat all procedures as if they were well-structured. You must get the degree of structure right in your analysis, and you must be aware of the degree of structure in your instruction.

Principle 8: Procedural Knowledge Uses Declarative Knowledge. Another common design error is to forget that there are always declarative knowledge components embedded in procedural knowledge—and that you must use appropriate instructional strategies to teach both.

Principle 9: People Solve Problems Inductively, But Only If It's an Unfamiliar Problem. Early work on problem solving sought to identify general (inductive) principles that could solve any problem. These turn out not to be used much by experts, and they usually are not the place to start when teaching novices.

Principle 10: An Expert Problem Solver Knows More and Different Things Than a Novice Does. An expert is not simply a novice after lots of practice. Experts know more, but they also know differently (with different knowledge structures). Therefore you need to understand the level of expertise of your learners and adapt your instruction accordingly.

Principle 11: Experts Know More Domain-Specific Strategies Than Novices Do. Experts know a great many patterns and insights (knowledge structures) specific to their problem domain. You should be looking for these in your cognitive task analysis (see Chapter Seven), and you should help the learner to see, learn, and use them.

Principle 12: Manage Cognitive Load in Training and in Performance. Cognitive load is all about how to prevent the cognitive ‘‘buffers’’ from ‘‘overflowing.’’ Yet management of cognitive load is still rare in instruction and assessment. See Chapters Nine and Ten for further discussion of how to manage cognitive load.

These principles are meant to help you see some of the most important implications for instructional design of current cognitive learning theory. As you study the chapters in this handbook, we hope these principles will help you build or revise *your* mental model of what learning is, what instruction is for, and how it works to help the learner do the work of learning.

References

Anderson, J. R. (1995). *Cognitive psychology and its implications* (5th ed.). New York: W.H. Freeman.

Anderson, J. R. (1995). *Learning and memory: An integrated approach*. Hoboken, NJ: John Wiley & Sons.

Dick, W., & Carey, L. (2001). *The systematic design of instruction* (5th ed.). New York: HarperCollins.

Ertmer, P. A., & Newby, T. J. (1993). Behaviorism, cognitivism, constructivism: Comparing critical features from an instructional design perspective. *Performance Improvement Quarterly*, *6*(4), 50–72.

Fleming, M., & Bednar, A. (1993). Concept-learning principles. In M. L. Fleming & W. H. Levie (Eds.), *Instructional message design: Principles from the behavioral and cognitive sciences* (2nd ed., pp. 233–252). Englewood Cliffs, NJ: Educational Technology Publications.

Foshay, R. (1991, May). Sharpen up your schemata. *Data Training*, pp. 18–25.

Gagné, R. (1985). *The conditions of learning* (4th ed.). New York: Holt, Rinehard and Winston.

Hannafin, M., & Hooper, S. (1993). Learning principles. In M. Fleming & H. Levie (Eds.), *Instructional message design* (2nd ed.). Englewood Cliffs, NJ: Educational Technology Publications.

Miller, G. A. (1956). The magical number seven, plus or minus two—some limits on our capacity for processing information. *Psychological Review*, *63*, 81–97.

Silber, K. H. (1998). The cognitive approach to training development: A practitioner’s assessment. *Educational Technology Research and Development*, *46*(4), 58–72.

West, C., Farmer, J., & Wolff, P. (1991). *Instructional design: Implications from cognitive science*. Boston: Allyn & Bacon.